PV204 Security technologies

Team projects

Petr Švenda
Faculty of Informatics, Masaryk University, Brno, CZ

CROCS
Centre for Research on Cryptography and Security

Project idea

Analyze and improve existing smartcard application

1. Select existing open-source JavaCard applet
2. Analyze the applet for security and performance
3. Improve the applet's code and add missing tests
4. Make applet ready for JavaCard Application Store
5. (Try to push changes to upstream repository)

Teams

- 3 people per team
  - Assigned today (within group), available in IS
- Teams must use GitHub for cooperation
  - Teams under JavaCardSpot-dev GitHub organization
  - Distribute work load evenly between all members
  - Contribution from all team members must be visible in git (git commits from each member)
  - Your evaluation will be partially based on your participation
- Teams may use existing code, but must make clear attribution to the original author

Basic hints on successful team work

- Form team from people with similar expectations
  - intended effort, final mark, interactions…
- Plan your work (GitHub milestones + issues)
- Don’t overcommit and fulfil your promises
- Agree on 4 personal session to work on project (at least 1 hour each) and block time in your calendar
  - Mail me the dates
- Every seminar 10 minutes reserved for team sync
  - Update your GitHub project milestones…

Projects – timeline (details on next slides)

1. Select target applet (1.3.2018)
   - No duplicate projects allowed
2. Fork & improve GitHub repository, compile&cap: 2 points (~~8.3.2018~~15.3.2018)
   - Under JavaCardSpot-dev organization, use gradle template
3. Perform security analysis: 5 points (22.3.2018)
   - Assert padding oracles, weak crypto, fault induction checks, cleaning of sensitive memory, storage of sensitive data unencrypted…
   - Report (max. 4 pages A4) + presentation (your seminar group)
4. Improve code (GitHub): 10 points (19.4.2018)
   - Best practices, functionality tests, version+repo info, state model…
5. Profile code performance: 3 points (3.5.2018)
   - Report + Presentation: Code improvements, performance results

- At least **10 points** (total) from the project are required

Gradle build JavaCard template

- https://github.com/ph4r05/javacard-gradle-template
- Pre-prepared JavaCard project template for building CAP and running JCardSim with gradle + coverage
  - IntelliJ Idea IDE project
  - Compilation and conversion of applet cap files
  - Support for easy tests creation including test coverage
  - Execution with real card or JCardSim.org simulator
  - Integration into Travis CI continuous integration
- Will be explained and used from the second week

How to work with existing repository

- Analyse if a project is 1) active or 2) abandoned
  - Analyse commit history (> 2 years since last commit?)
  - Try to contact authors (email, message on GitHub)

1. If actively maintained project
   - Fork target repo under JavaCardSpot-dev organization
     - Only small changes will be done + pull request
   - Create second repository *original_name-build*
     - This repo will contain only gradle build chain, tests…
2. If abandoned project
   - Fork original repo JavaCardSpot-dev organization
   - add gradle build + tests directly

Project: Extending existing repository

- Your repo is already forked under JavaCardSpot-dev with team access rights assigned
- Add README.md (if missing) + notice + details
- Setup with gradle template (details on next slide)
- Remove all compilation and conversion warnings and errors (cap file)
- Make working with TravisCI (no tests required yet, just compile and convert)
  - Add TravisCI badge [![Build status](https://travis-ci.org/JavaCardSpot-dev/yourproject.svg?branch=master)](https://travis-ci.org/JavaCardSpot-dev/yourproject)

Project: Extending existing repository

- Read https://github.com/crocs-muni/javacard-gradle-template-edu/blob/master/INTEGRATION.md (thanks to Dusan Klinec)
- We will start with Variant A (easiest)
  1. Your repository is already forked under JavaCardSpot-dev with team access rights assigned
  2. Clone this repo to your local repo on your machine (git or GitHub app)
  3. Download ZIP with gradle template: https://github.com/crocs-muni/javacard-gradle-template-edu/archive/master.zip
  4. Copy template from ZIP to your local repo and move original applet source code to applet/src/main/java directory
  5. Try and fix compilation, cap conversion and test execution
- (We will later cherry-pick changes for pull request to orig repo)
- Optional: If your team is skilled, feel free to pick other more suitable variant directly

Project : Security analysis

- Report (max. 4 pages A4) + presentation (22.3., your seminar group)
  - What functionality is offered + APDU format
  - What sensitive values are protected
  - What cryptographic algorithms and protocols are used
  - What is relevant attacker model
  - What is state model of applet
- Best security practices
  - padding oracles, weak crypto, fault induction checks, cleaning of sensitive memory, storage of sensitive data unencrypted…

PROJECTS AVAILABLE FOR SELECTION

- OpenPGP applet [Active]
  - https://github.com/JavaCardSpot-dev/ykneo-openpgp
  - *Jan Masarik, Šimon Struk, Svetlana Viktória Stuchlá*
- SmartPGP applet [Active]
  - https://github.com/JavaCardSpot-dev/SmartPGP
  - *Marek Vančík, MarekVan, Peter Benčík, Jakub Martinka*
- PKI Windows login applet
  - https://github.com/JavaCardSpot-dev/GidsApplet
  - *Kuldeep Goyal, Deniz Agaoglu, Loic Nicolas*
- U2F NFC authentication applet [Active]
  - https://github.com/JavaCardSpot-dev/ledger-u2f-javacard
  - *Jan Jancar, Pavel Brousek*
- EMV payment applet (move to GitHub)
  - https://sourceforge.net/projects/javacard-openemv-applet
  - *Chintan Khanna, Gajraj Kuldeep, Niraj Kalra*

- Key manager for Cryptsetup [Active]
  - https://github.com/JavaCardSpot-dev/cryptsetup-javacard
  - *Urvekkumar C Shah, Hitesh Lilhare*
- KeePass NFC applet
  - https://github.com/JavaCardSpot-dev/KeePassNFC
  - *Akhilesh Soni, Marco Ciotola, Vikas Lamba*
- Ledger Bitcoin wallet [Active]
  - https://github.com/JavaCardSpot-dev/ledger-javacard
- SatoChipApplet Bitcoin wallet
  - https://github.com/JavaCardSpot-dev/SatoChipApplet
  - *Martin Knotek, Lenka Svetlovska, Jiri Tyma*
- Software cryptographic primitives for JavaCard
  - https://github.com/JavaCardSpot-dev/Primitives_SmartCard
  - https://github.com/petrs/JCSWAlgs
  - *Arvind Rao, Bhupendra Singh, Ram Singh*

Projects available for selection

- OpenTLS implementation
  - https://github.com/JavaCardSpot-dev/opentlssc
- Client implementation of TLS
  - https://github.com/JavaCardSpot-dev/smart_card_TLS
  - *Sujeet Deshmukh, Nidhi Pokhriyal, Surendra Kumar Yadav*
- PIV applet FIPS201 [Active]
  - https://github.com/JavaCardSpot-dev/OpenFIPS201
- *Or other project from list (must agree with me)*
  - https://github.com/EnigmaBridge/javacard-curated-list